I HEREBY CERTIFY that annexed hereto is a true copy of documents filed in connection with the following patent application:

Application No.   S2000/0549

Date of Filing   6 July 2000

Applicant   RICHMOUNT COMPUTERS LIMITED, an Irish
                       Company of Maple House, South County Business
                       Park, Leopardstown, Dublin 18, Ireland.

Dated this 16   day of March 2001.

Corcoran

An officer authorised by the
Controller of Patents, Designs and Trademarks.

FORM NO. 1

REQUEST FOR THE GRANT OF A PATENT
PATENTS ACT, 1992

The Applicant named herein hereby request

__ the grant of a patent under Part II of the Act

X the grant of a short-term patent under Part III of the Act

on the basis of the information furnished hereunder.

1. **APPLICANT**

   Name   **RICHMOUNT COMPUTERS LIMITED**

   Address   **Maple House, South County Business Park, Leopardstown, Dublin 18, Ireland.**

   Description/Nationality   **An Irish Company**

2. **TITLE OF INVENTION**

   "Fibre Channel Diagnostics in an Enclosure Without Using a Switch"

3. **DECLARATION OF PRIORITY ON BASIS OF PREVIOUSLY FILED APPLICATION FOR SAME INVENTION (SECTIONS 25 & 26)**

Previous filing date	Country in or for which filed	Filing No.

4. **IDENTIFICATION OF INVENTOR(S)**
   Name(s) of person(s) believed by Applicant(s) to be the inventor(s)
   1.

   Address
   1.

5. **STATEMENT OF RIGHT TO BE GRANTED A PATENT (SECTION 17(2)(B))**

   By virtue of

   Contd./...

6. **ITEMS ACCOMPANYING THIS REQUEST - TICK AS APPROPRIATE**

(i)   X prescribed filing fee EUR 63.49 (IR£50.00)

(ii) _ specification containing a description and claims

   X specification containing a description only

   X Drawings referred to in description or claims

(iii) _ An abstract

(iv) _ Copy of previous application(s) whose priority is claimed

(v)   __ Translation of previous application whose priority is claimed

(vi)   __   Authorisation of Agent (this may be given at 8 below if this Request is signed by the Applicant(s))

7. **DIVISIONAL APPLICATION**
   The following information is applicable to the present application which is made under Section 24

   Earlier Application No:
   Filing Date:

8. **AGENT**

   The following is authorised to act as agent in all proceedings connected with the obtaining of a Patent to which this request relates and in relation to any patent granted -

   Name   **Address**
   F. R. KELLY & CO.   at their address as recorded for the time being in the Register of Patent Agents

9. **ADDRESS FOR SERVICE (IF DIFFERENT FROM THAT AT 8)**
   RICHMOUNT COMPUTERS LIMITED
   F. R. KELLY & CO.

   By: _____
                    EXECUTIVE

Date: July 6, 2000

FIBRE CHANNEL DIAGNOSTICS IN AN ENCLOSURE WITHOUT USING A SWITCH

Overview

There are currently three enclosure hub architectures under consideration , the XL500 type architecture, the VSC7148 type architecture and a digital hub type architecture (Vixel, Gadzooks). This paper introduces another alternative for discussion. The architecture comes in two parts, a Loop Data Integrity Checker and a hub based on an analogue crosspoint switch.

Nomenclature

There is plenty of confusion about what constitutes a hub and a switch, in order to clarify that I've defined them here:

Hub: A hub is really a repeater which works by exposing all possible destinations to a packet from a given source (and the interested destination takes it).

Switch: A switch is something that makes use of the datalink addresses to move traffic only between the interested source and destination pairs.

Crosspoint Switch: A device with several inputs and outputs which can allow any output to be connected to any input. Also known as a crossbar switch. Despite it's name it is not a switch as defined above, although it may be used as one component in a switch.

In a lot of cases we say switch when we really mean hub. All the proposals on the table at the moment are hub based proposals, not switch based. It's possible the Gadzooks chip may be a hybrid (hub for some types of data frames, switch for others) but that is not clear at the time of writing.

Problems

The problems we are trying to solve at the moment are:

- We want to add a significant amount of loop diagnostics capability.
- We do not want to lower performance significantly.
- We do not want a huge increase in product cost.

Loop Data Integrity Checker

A Loop Data Integrity Checker (LDIC) consists of a SerDes such as a Vitesse 7126 and an FPGA. The SerDes is connected in receive only mode to an FC-AL loop, it is not a member of the loop and thus does not contribute to the Loop Delay. The FPGA receives the data from the SerDes and performs basic checking on it, including checking the CRC of every frame. A summary of the data analysed by the FPGA is passed on to the SES processor. A block diagram of an LDIC is shown in Figure 1 below:
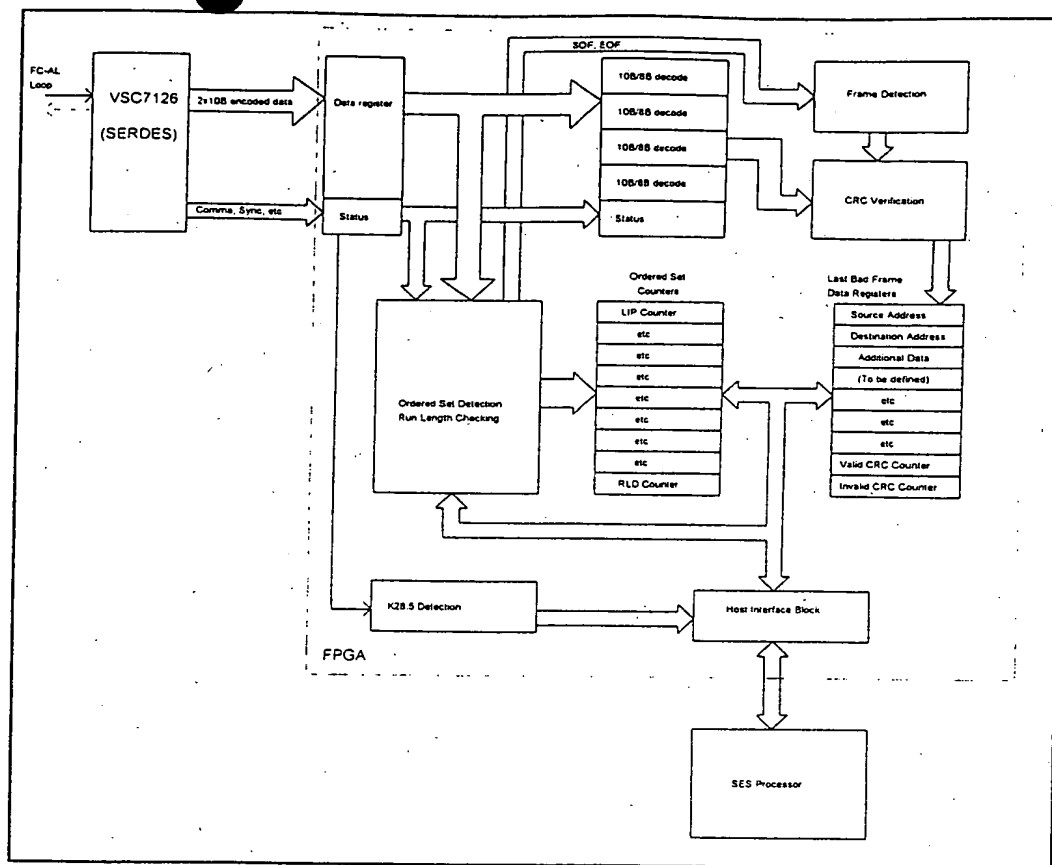
1/4

[D006]

Figure 1: Loop Data Integrity Checker

Features include:

- Can detects and count LIPS of any variety.
- FPGA supports the detection of several common types of ordered set, plus one or two versatile checkers which can detect any required ordered set. The versatile checkers are programmable from the SES processor.
- Detects loss of sync, reports no of times sync lost and how long it was lost for in total.
- Detects and counts run length disparities.
- Detects missing K28.5 character, ie K28.5 not received in the appropriate time.
- Detects and counts bad frame CRCs and records the source and destination of one bad frame. The source and destination data remains static until it has been read by the SES processor, the SES processor then re-arms the circuitry for the next bad frame.
- Counts frames with valid CRSs and frames with invalid CRCs.
- All counters can be reset to zero by the SES processor, there is no adverse effect on the system if they are not regularly read or cleared.
- The FPGA configuration is downloaded from the SES processor to allow on line upgrades to the FPGA.
- Can be enhanced to generate simple data structures and primitives.
- Can be enhanced to move "up the stack" and analyse some basic SCSI performance metrics, similar to those in Mike Lardner's white paper on SCSI Performance Monitoring.
- Can be enhanced to measure the loop delay.

2/4

Conclusions

- A high level of diagnostics capability can be achieved without a digital hub chip.
- The current cost of a crosspoint hub is comparable with a digital hub. However there are a lot of opportunities for cost reduction of the crosspoint version.

- I feel this architecture deserves further discussion investigation.

314

What is the problem to be solved by this invention or what is the need for this invention? :

i.e. Describe the problem and/or explain the need clearly.

When a problem occurs on a Fibre Channel loop it is extremely difficult to work out what the problem is and what device on the loop is causing the problem. This invention Assists in that task, particularly when used in conjunction with an analogue cross point switch (see separate patent disclosure)

What are the closest known technologies, products or processes to the invention? :
i.e. Are there alternative ways of solving the problem?

Loop diagnostic information can be obtained by replacing the analogue port bypass circuits in the loop with a digital switch or hub, with diagnostic hardware on every port.

What are the problems with closest known technologies, products or processes? :
i.e. Why don't they solve the problem, or what are their disadvantages? - e.g. cost, quality, cycle time, repeatability, manufacturability.

Digital hubs are expensive and seriously degrade performance.
Digital switches are very expensive, they increase performance in a single shelf configuration, they may increase performance in a multi-shelf situation.

Explain how this invention overcomes these problems:
i.e. What are the advantages of this invention compared to known technologies, products or processes?

This invention does not require the use of a digital switch or hub chip and thus does not affect performance. It is also relatively cheap.

What are the possible applications for this invention? :
i.e. List applications, including opportunities for future products.

It could potentially be used in any fibre channel shelf.

What is the perceived commercial value of the invention :
i.e. cost savings per unit or increased revenue per annum.
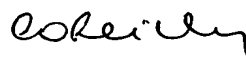
4/4

I HEREBY CERTIFY that annexed hereto is a true copy of documents filed in connection with the following patent application:

Application No.   S2000/0710

Date of Filing   7 September 2000

Applicant   RICHMOUNT COMPUTERS LIMITED, an Irish Company of Maple House, South County Business Park, Leopardstown, Dublin 18, Ireland.

Dated this 16 day of March 2001.

CoReilly

PP An officer authorised by the
Controller of Patents, Designs and Trademarks.

FORM NO. 1

REQUEST FOR THE GRANT OF A PATENT
PATENTS ACT, 1992

The Applicant named herein hereby request

___ the grant of a patent under Part II of the Act

__X__ the grant of a short-term patent under Part III of the Act

on the basis of the information furnished hereunder.

1. **APPLICANT**

   Name   **Richmount Computers Limited**

   Address   **Maple House, South County Business Park, Leopardstown, Dublin 18, Ireland**

   Description/Nationality   **An Irish company**

2. **TITLE OF INVENTION**

   "Fibre Channel Diagnostics in a Storage Enclosure"

3. **DECLARATION OF PRIORITY ON BASIS OF PREVIOUSLY FILED APPLICATION FOR SAME INVENTION (SECTIONS 25 & 26)**

   Previous filing date   Country in or for which filed   Filing No.

4. **IDENTIFICATION OF INVENTOR(S)**
   Name(s) of person(s) believed by Applicant(s) to be the inventor(s)
   1. Aedan Coffey

   Address
   1.

5. **STATEMENT OF RIGHT TO BE GRANTED A PATENT (SECTION 17(2)(B))**

   By virtue of

   Contd./...

6. **ITEMS ACCOMPANYING THIS REQUEST - TICK AS APPROPRIATE**

    (i)   _X_ prescribed filing fee EUR 63.49 (IR£50.00)

    (ii) _ specification containing a description and claims

          _X_ specification containing a description only

          _X_ Drawings referred to in description or claims

    (iii) _ An abstract

    (iv) _ Copy of previous application(s) whose priority is claimed

    (v)   _ Translation of previous application whose priority is claimed

    (vi) _ Authorisation of Agent (this may be given at 8 below if this Request is signed by the Applicant(s))

7. **DIVISIONAL APPLICATION**
    The following information is applicable to the present application which is made under Section 24

    Earlier Application No:
    Filing Date:

8. **AGENT**

    The following is authorised to act as agent in all proceedings connected with the obtaining of a Patent to which this request relates and in relation to any patent granted -

    Name   **Address**
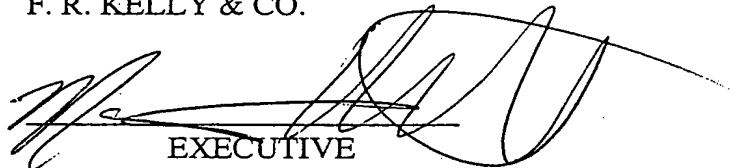   F. R. KELLY & CO.   at their address as recorded for the time being in the Register of Patent Agents

9. **ADDRESS FOR SERVICE (IF DIFFERENT FROM THAT AT 8)**
    RICHMOUNT COMPUTERS LIMITED
    F. R. KELLY & CO.

                                    By: EXECUTIVE

    Date: September 7, 2000

    0 7 SEP 2000

S 000710

Fibre Channel Diagnostics in a Storage Enclosure

Field of Invention

5   This invention relates to an analyser of the performance of a
bus included within a storage enclosure.

Related Applications

10   The invention herein disclosed is related to co-pending
application number _____filed on even date entitled
"Cross-Point Switch for a Fibre Channel Arbitrated Loop"
naming Aedan Diarmid Cailean Coffey as inventor (Attorney
docket number PI 29278); to co-pending application number
15   filed on even date entitled "A Data Gathering Device for a
Rack Enclosure" naming Aedan Diarmid Cailean Coffey as
inventor (Attorney docket number PI29273); and to co-pending
application number _____ filed on even date entitled
"Performance Monitoring in a Storage Enclosure"   naming Aedan
20   Diarmid Cailean Coffey as inventor (Attorney docket number
PI29276).

Background of Invention

25   Performance improvements in storage and processors, along with
the move to distributed architectures such as client/server
systems, have spawned increasingly data-intensive and high-
speed networking applications, such as multimedia and
scientific visualisation. Such applications have placed
30   growing demands of the performance on the interconnects
between host computers and input/output devices in terms of
their reliability, speed and distance.

Fibre Channel (FC) is a general name for an integrated set of
standards being developed by ANSI (American National Standards
Institute) whose purpose is to act as a universal high-speed
interface for computers and mass storage. It is designed to
5   combine the best features of channels and networks, namely the
simplicity and speed of channel communications and the
flexibility and interconnectivity of protocol-based network
communications. FC is a data transfer protocol that provides
a highly reliable, gigabit interconnect technology that allows
10   concurrent communications among workstations, mainframes,
servers, data storage systems and other peripherals using
Small Computer Systems Interface (SCSI) and Internet Protocol
(IP) protocols. FC supports multiple topologies, including a
Fibre Channel Arbitrated Loop (FC-AL), which can scale to a
15   total system bandwidth on the order of a terabit per second.
However, system performance limitations may be introduced as a
result of inefficient system configuration, e.g., where a
legacy device on a network bus determines the overall bus
speed. In such situations, it is clearly of benefit for a
20   network analyst to be able to monitor the performance of the
network and optimise its configuration and/or diagnose faults.

When a problem occurs on a Fibre Channel Arbitrated Loop (FC-
AL) it can be extremely difficult to determine the nature of
25   the problem and identify which device on the loop is causing
the problem. This is the case because from a logical point of
view, an arbitrated loop is a single, continuous path composed
of links and nodes, wherein each node has at least one port
which can act as a transmitter, receiver or both. Hence it
30   can be difficult to identify the specific node involved in a
device failure since there may be no obvious indication of the
location of the failure point in the loop.

Conventional analysers of a Fibre Channel Arbitrated Loop (FC-AL) performance are large and expensive stand-alone devices, which are usually connected to a FC-AL, only when it is suspected that a problem exists thereon. Such stand-alone FC-AL analysers provide very detailed analyses of bus traffic, in addition to a wide range of user-selectable capture modes and triggering options.

Disclosure of Invention

The present invention provides a fibre channel analyser for analysing the operation of a fibre channel arbitrated loop to which a plurality of devices are connectable, said analyser being adapted to be housed in an enclosure which, in use, houses at least one of said devices and comprising:

means for extracting data from the fibre channel,

means for processing extracted data; and

means for communicating processed data to an environmental control and monitoring unit through a secondary communication bus.

The invention extends the functionality of the system environmental control and monitoring unit to encompass an analysis of the fibre channel itself, and by using the host communication facilities of the monitoring unit, the footprint of the analyser can be made such that it can be housed within the enclosure.

Preferably, the analyser is sufficiently small and inexpensive to be included directly within a FC-AL enclosure thereby enabling continuous on-line monitoring of the FC-AL bus and the provision of an early warning system of FC-AL bus performance degradation.

Preferably, the analyser does not provide as detailed an analysis of the FC-AL bus performance as a conventional stand-alone FC-AL analyser, however the invention does provide

5   sufficient information to enable a network analyst to perform a status check of the system. Further, the analyser preferably also provides information on the transmission of both ARB (Arbitrate) and LIP (Loop Initialisation) ordered sets. This is important because, the presence of a LIP on an

10 FC-AL can indicate that a new loop port has been added to the loop, a loop failure has been detected, or a port suspects that another port on the loop may be hung. Further, the LIP emitting loop port may be unable to co-ordinate transmission of the LIP sequence with current loop traffic. If the

15 initialising loop port begins transmission of LIP while frames are being sent on the loop, it is possible that one or more of the frames may be corrupted.

Preferably, the analyser arranged to be located on one of a

20 number of branches from the FC-AL and not in the loop itself. Disks and hosts are located in the FC-AL wherein data is actively repeated from one node on the FC-AL to another, with a subsequent delay arising from the repeating process associated with each node. Further, if one node on the loop

25 fails then the entire transmission process on the rest of the loop also fails. The preferred embodiment analyses activity occurring on the loop but does not itself contribute to loop delay. Further the branching connection structure employed with the analyser means that a failure in the analyser will

30 not cause the rest of the loop to fail.

Preferably, the analyser comprises one or two chips as opposed to the multiple chips and cards in existing stand-alone

systems and analysers.

Preferably, the monitoring unit comprises an Enclosure
Services processor communicating with the bus controller by
5   methods including the SCSI Enclosure Services (SES) or SCSI
Access Fault Tolerant Enclosure (SAF-TE) protocols. The SCSI
Enclosure Services (SES) processor includes facilities for FC-
AL bus monitoring by the addition of:

•   a control page to enable a user to specify the levels of
10   analysis of bus performance required; and

•   a status page containing processed data results from the
analysis performed by the bus analyser.

Brief Description of the Drawings

15

The features and advantages of the present invention will
become apparent from the following description of the
invention, taken together with the accompanying drawings, in
which:

20

Figure 1 is a block diagram showing a broad overview of a
Fibre Channel Arbitrated Loop (FC-AL) Analyser and the manner
in which it relates to other elements in an integrated data
gathering system for a Fibre Channel Arbitrated Loop (FC-AL);

25

Figure 2 is a block diagram of the components of a frame;

Figure 3 is a block diagram of the components of a Fibre
Channel Arbitrated Loop (FC-AL);

30

Figure 4 is a diagram showing a broad overview of an example
scenario showing how a Fibre Channel Arbitrated Loop (FC-AL)
analyser might be used on a Fibre Channel Arbitrated Loop;

Figure 5 is a more detailed diagram of the cross-point switch illustrated in Figure 1 (A specific example of the manner in which it might be used is made with reference to the example scenario shown in figure 4); and

5

Figure 6 is a detailed block diagram of the FC-AL analyser of Figure 1.

Description of the Preferred Embodiment

10   Figure 1 is a block diagram showing a broad overview of a Fibre Channel Arbitrated Loop (FC-AL) and the manner in which it relates to other elements in an integrated data gathering system for the FC-AL. The overall operation of this system is described in co-pending application entitled "Data Gathering

15   Device for a Rack Enclosure" naming Aedan Diarmid Cailean Coffey as inventor (Attorney docket number PI 29273).

In the preferred embodiment, a plurality of disks (80) are housed in a rack and engage a back-plane (10) within the rack

20   through edge-connectors (not shown). The disks are electrically and logically connected to form two FC-AL Loops A and B via respective hubs, each comprising a cross-point switch (also known as cross-bar switch) (30,30') as described in related patent application number entitled "Cross-Point

25   Switch for a Fibre Channel Arbitrated Loop" naming Aedan Diarmid Cailean Coffey as inventor. The disks are then in turn configured to form a redundant array of independent disks (RAID) or Just a Bunch of Disks (JBOD).

30   A pair of FC-AL analysers (70,70') are preferably located on each of pair of SCSI Enclosure Services (SES) processor boards (4,4'), with each board being associated with one of Loops A or B. Nonetheless, the analysers may also be located on the

back-plane (10) or wherever else in the integrated data
gathering system as would enable the analyser to be on the FC-
AL. The operation of both SES processor boards (4,4') is
identical and so reference will only be made in the present

5   description to the board 4 and its components.

Again, the positioning of the analysers on the FC-AL is
managed by the cross-point switch (30), however, it should be
recognised that the analysers could also operate via a

10   modified conventional type port-bypass circuit by sacrificing
the benefits of using the cross-point switch, as explained
below.

According to a preferred embodiment of the invention, the FC-

15   AL analyser (70) serves as an adjunct to the SES Processor
(7). The SES Processor (7) of the preferred embodiment
provides online monitoring and control of variables
predominantly associated with the enclosure environment (e.g.
temperatures at different locations in the enclosure, fan

20   speed, power supply voltages and currents and presence/absence
of I/O controls, loop relay circuits and device drivers).
Further details of the SES Processor (7) can be obtained in
co-pending patent application entitled "Performance Monitoring
in a Storage Enclosure" naming Aedan Diarmid Cailean Coffey as

25   inventor.

In figure 1 it can be seen that the SES processor (7) is in
bi-directional communication with disks (80) on the FC-AL, via
one of a pair of Data Gatherer Chips (50,50') through an

30   Serial Peripheral Interface (SPI) bus (54) and an Enclosure
Services Interface (ESI) bus (52) (also known as Small Form
Factor SFF-8067). (Where data gatherer chips are not employed,

the SES processor can connect directly to the ESI ports of the disks.)

Through communication between components of the FC-AL itself,
5   communications from the SES processor (7) to FC-AL disks (80)
are transmitted to a Host CPU (not shown) on the FC-AL.
Further references to communication between the SES processor
(7) and a host CPU will assume communication through the Data
Gatherer Chip (50) and FC-AL disks (80) and will assume that
10 the host CPU is a node on the FC-AL itself.

Since this operation of the analyser (70) involves the
detection of transmission errors on a FC-AL, it is useful at
this point to briefly review fibre channel (FC) transmission
15 protocols, the FC-AL topology and the types of errors that
occur in such systems.

The Open Systems Interconnection (OSI) model for FC is
structured with 5 independent layers as follows;
20

- FC-0 which defines the physical media and transmission rates
- FC-1 which defines the transmission protocol including
serial encoding and decoding rules, special characters,
timing recovery and error control.
25 - FC-2 which defines the framing protocol and flow control
- FC-3 which defines the common services
- FC-4 which defines the application interfaces that can
execute over FC such as SCSI, IPI and IP.

30 From this it can be seen that the FC protocol does not have
its own command set, but merely manages the data transfer
between participating devices and thus inter-operates with

existing upper-level protocols such as Small Computer System
Interface (SCSI-3), Intelligent Peripheral Interface (IPI) and
Internet Protocol (IP). Hence a complete analysis of a FC-AL
could include a higher-level analysis of the SCSI protocol
5   commands issued on the FC-AL network in addition to the lower-
level analysis of the FC protocol.

Hence, the analyser 70 not solely limited to the analysis of
FC characters, but can also be extended to include the
10   analysis of SCSI commands on the FC-AL, by integrating the
functionality of SCSI analyser (5) as described in related
patent application number entitled "Performance Monitoring in
a Storage Enclosure" naming Aedan Diarmid Cailean Coffey as
inventor (Attorney docket number PI 29276), with that of the
15   FC-AL analyser (70).

Fibre Channel (FC) Components

Devices that can be accessed via FC are known as *nodes*. FC
20   nodes have at least one port (known as an N-port) such ports
can act as transmitters, receivers or both. The term NL_port
is used to designate a N_port that can support arbitrated loop
functions in addition to basic point-to-point functions. A
node that initiates a transaction is known as an *originator*,
25   the node that answers it is called a *responder*.

Fibre Channel (FC) Transmission Protocols

Before it is transmitted every byte of data is encoded into a
30   10 bit string known as a *transmission character* (using an
8B/10B encoding technique (Patent No. US 4486739)). Each un-
encoded byte is accompanied by a control variable of value D
or K, designating the status of the rest of the bytes in the

transmission character as that of a data character or a
special character respectively.

The encoding from an 8-bit data byte into a 10-bit code is
5 achieved according to an 8B/10B-translation table and a
running disparity calculated from a bit-stream. The *running
disparity* is calculated as the number of ones minus the number
of zeros sent in the bit-stream and is proportional to the DC
level of the bit-stream. The 8B/10B-translation table
10 includes two entries, corresponding to a positively or
negatively valued *running disparity* for each 8-bit data byte.
The entry is chosen to keep the *running disparity* for a given
8-bit data byte between +1 and -1 so that the DC balance is
maintained near zero.
15
In general, the purpose of this encoding process is to ensure
that there are sufficient transitions in the serial bit-stream
to make clock recovery possible. The *8B/10B encoding
technique* supplies sufficient error detection and correction
20 to permit use of low cost transceivers, as well as timing
recovery methods to reduce the risk of radio frequency
interference and ensure balanced, synchronised transmissions.

Whilst, every 8-bit data byte is encoded as a 10 bit
25 transmission character according to this encoding process,
there are however, many more possible 10 bit *transmission
characters* than are needed to map to particular 8-bit data
bytes. Only one of the remaining 10 bit encodings is of
interest in this present description, namely the *K28.5
30 transmission character*. This character contains a "comma", a
7-bit string that cannot occur in any *data transmission
character* (i.e. a *transmission character* corresponding to a

data character) because of this, the K28.5 is used as a
special control character.

As discussed above, the *8B/10B encoding technique* provides a
means of synchronisation to a received signal, however it also
provides a means for error detection. *Invalid transmission
characters* are *transmission characters* that have not been
defined according to the 8B/10B-translation table. *Invalid
transmission characters* also includes *those transmission
characters* that are received or transmitted with an incorrect
running disparity.

All information in FC is transmitted in groups of four
transmission characters called *transmission words* (40 bits).
Some *transmission words* have the *K28.5 transmission character*
as their first *transmission character* and are called *ordered
sets*. *Ordered sets* provide a synchronisation facility which
complements the synchronisation facility provided by the
8B/10B encoding technique. Whilst phase locked loops (PLLs)
enable synchronisation on the bit level with the assistance of
the 8B/10B encoding technique, the responder also needs to
synchronise with the originator at the 40 bit level. *Ordered
sets* provide for both bit and word synchronisation. Such
synchronisation establishes word boundary alignment, since the
K28.5 transmission character can not be transmitted across the
boundaries of any two adjacent *ordered sets* unless an error
has occurred. Synchronisation is deemed to have occurred when
the responder identifies the same *transmission word* boundary
on the received bit-stream as that established by the
originator.

An *ordered set* may be a *frame delimiter*, a *primitive signal* or
a *primitive sequence*. A *frame delimiter* includes one of a

Start_of_Frame (SOF) or an *End_of_Frame (EOF)*. These *ordered sets* immediately precede or follow the contents of a *frame*, their purpose is to mark the beginning and end of *frames*. *Frames* will be discussed in more detail below. *Primitive*

5   *signals* are normally used to indicate events or actions. The set of *primitive signals* is comprised of the *Idle* and *Receiver Ready (R_RDY)* ordered sets. An *Idle* is a primitive signal transmitted continuously over the link when no data is being transmitted. The *Idle* is transmitted to maintain an active

10   link over a fibre and enables the responder and originator to maintain bit, byte and word synchronisation. The *R_RDY primitive signal* indicates that an interface buffer is available for receiving further frames. *Primitive sequences* are used to indicate states or conditions and are normally

15   transmitted continuously until something causes the current state to change. Such sequences include *Offline (OLS)*, *Not Operational (NOS)*, *Link Reset (LR)* and *Link Reset Response (LRR)*, all of which are used in the process of initialising a link between two N-ports

20

A frame is the smallest indivisible packet of information transmitted between two N_Ports. Figure 2 shows a diagrammatic representation of a frame. A frame (110) is comprised of a *Start_of_Frame (SOF) ordered set* (112), a

25   header (114), a payload (116), the *Cyclic Redundancy Check (CRC)* (118) and an *End_of_Frame (EOF) ordered set* (120). The header (114) contains information about the frame, including routing information (the source and destination addresses (122 and 124), the type of information contained in the payload

30   ·(126) and sequence exchange/management information (128).

The payload (116) contains the actual data to be transmitted and can be of variable length between the limits of 0 and 2112

bytes. The CRC (118) is a 4-byte record used for detecting bit errors in the frame when received. The total size of a *frame* can be variable but must be an even multiple of four bytes so that partial *transmission words* are not sent.

5 Individual *frame* sizes are transparent to software using the FC because the groups of one or more related *frames* responsible for a single operation are transmitted as a unit, such units being known as *sequences*.

Fibre Channel Arbitrated Loop (FC-AL)

10

FC-AL is a loop interconnection topology that allows up to 127 participating node ports (one of which can be a fabric loop port providing attachment to a switched fabric) to communicate with each other without the need for a separate switched

15 fabric. Instead of a centralised approach to routing, the FC-AL distributes the routing function to each loop port.

Figure 3 shows a diagrammatic representation of a four node FC-AL. The FC-AL comprises four nodes (130, 131, 132 and 133)

20 connected together via their ports (134, 135, 136 and 137). Information flows between the ports in a unidirectional fashion.

The arbitrated loop configuration is created by connecting a

25 transmit output section of each port to a receive input section of the next loop port (e.g. connecting the transmit output section of Node 1 Port 1 (139) to the receive input section of Node 2 Port 2 (140)). Signal transmission continues through the remaining nodes on the FC-AL, until the

30 signal reaches its designated responder. In other words, information from a given port (i.e. the originator) flows around the loop to its designated responder through each of the intermediate ports. Each port on the loop contains a

repeater (146, 147, 148 and 149) allowing *frames* and *ordered sets* to pass through the port.

Loop-specific protocols are defined to control loop
5   initialisation, arbitration and the opening and closing of
loop circuits. These protocols use *primitive signals* and
primitive sequences comprised of *loop-specific ordered sets*.
The *loop-specific ordered sets* act as an addendum to those
ordered sets previously defined by the Fibre Channel Standard,
10 which have been specifically developed to implement the FC-AL
protocols.

FC-AL does not add any new frame delimiter *ordered sets*.
Additional Primitive Signals include those for arbitration
15 (e.g. *ARBx*), clock synchronisation (e.g. *SYNx*), and opening
(e.g. *OPNy*) and closing (*CLS*) communications between specific
nodes. Additional Primitive Sequences include those for loop
initialisation (*LIP*) and loop port bypass and enablement.

20 **Operation of the FC-AL**

Loop initialisation is used to initialise the loop, assign
addresses to the ports on the loops, known as *Arbitrated Loop
Physical Address (AL_PA)*, and provide notification that the
25 configuration may have changed. Loop initialisation is
achieved by means of the *Loop Initialisation Primitive (LIP)
sequence* and a series of loop initialisation *frames*. Any loop
port on the loop is capable of starting loop initialisation by
entering the initialising state and transmitting one of the
30 LIP sequences.

The loop is a common resource shared by all loop ports. In
order to ensure that information from one loop port does not

interfere with information from another, each loop port must arbitrate for access to the loop and win arbitration before they transmit frames of their own on the loop. When a device is ready to transmit data, it arbitrates for access to the

5   loop by transmitting the Arbitrate (ARBx) Primitive Signal, where x= the *Arbitrated Loop Physical Address (AL_PA)* of the device, which it then transmits to the next node in the loop.

If no other device wishes to transmit, the *ARBx* is transmitted

10   around the loop through each node in turn, until it returns to the original arbitrating node. Once the node has received its own *ARBx* Primitive Signal it has gained control of the loop.

However, if more than one device on the loop is arbitrating at

15   the same time, when an arbitrating device receives another device's *ARBx*, it compares the x value of the received *ARBx* (i.e. the *AL_PA* of the originator) with its own *AL_PA*. The device transmits the *ARBx* with the numerically lower *AL_PA* while the *ARBx* with the numerically larger *AL_PA* is blocked.

20   Thus the device with the lower *AL_PA* will gain control of the loop first. Once that device relinquishes control of the loop, the other device will have another chance at arbitrating for control.

25   After a loop port has won arbitration (and hence has become an originator), it must then select a destination port (or a responder port) before sending frames to that port. This selection process is known as opening the destination port and uses the open (*OPN*) ordered set that the originator transmits

30   to the responder. Once this happens, there essentially exists a point to point connection between the two devices. Only the originator and responder ports in the loop circuit are able to originate frame transmission. All the other devices in the

loop between the originator and the responder device simply
repeat the data.

As long as the loop circuit is active, the originator and
5   responder ports have full use of the loop's bandwidth. Each
loop may simultaneously transmit and receive data. When the
two ports have completed communication with each other, the
circuit is closed and the loop is made available for use by
other ports.

10

FC-AL Errors

(1) Link Errors

15   Link errors can occur during the transmission of the ordered
sets used to implement the loop protocols. Most link errors
will result in an 8B/10B error manifested as either an *invalid
transmission character* or running disparity error. Some link
errors may result in a valid but incorrect *transmission
20   character* being decoded. If a node on an FC-AL receives an
invalid transmission character while in the monitoring or
arbitrating states, it substitutes any valid character for the
invalid transmission character in order to create a valid
word.

25

This behaviour introduces the possibility that a node could
detect an *invalid transmission character* of an ordered set
destined for another node and replace any *transmission
character* in the *ordered set* with a different one. The
30   substitution may result in an *ordered set* being unrecognisable
by the receiving node. There is also the possibility that an
ordered set could be transformed into a different valid
ordered set, or that an AL_PA value in the *ordered set* could

be transformed into a different AL_PA value.

If an *ordered set* is corrupted and unrecognisable, the action taken by a receiving node depends on the current state of the node. An invalid *ordered set* can be either retransmitted, discarded with the port continuing normal transmission with an appropriate fill word or another *ordered set*.

(2) Loop Protocol Errors

Loop Protocol Errors can occur as a result of lost *ordered sets*, incorrect *ordered sets* or unexpected *ordered sets*. A lost *ordered set* is one that is never recognised by its intended recipient. It could have been corrupted by a link error or due to a failure in the sending port. Lost *ordered sets* result in an expected action never occurring, an incorrect action occurring or the action occurring at the wrong port.

The errors can occur during any of the loop protocols, including arbitration, while opening or closing a loop circuit and during frame transmission or initialisation. For instance errors during the arbitration protocol may result in one or more ARB primitives being lost or corrupted.

(3) Other Errors

In addition to the errors that are unique to the FC-AL topology, the FC-AL environment is also subject to all of the normal errors that can occur in non-loop environments. A *frame* may be lost or misrouted if it is delivered to the wrong port or if the SOF delimiter is corrupted. A *frame* may contain a CRC error. A *frame* may also contain an invalid

transmission word. An invalid transmission word is recognised by the responder when one of the following conditions is detected; an invalid transmission character is detected within a *transmission word* (in accordance with the 8B/10B-translation table), a special character alignment error is detected.
5 (e.g., a K28.5 character is received as an odd-numbered character). In addition, errors can affect the flow control mechanisms using R_RDY and ACK ordered sets.

10 Turning now to the FC-AL analyser, figure 4 depicts a broad overview of an example scenario showing how the FC-AL analyser might be used on a Fibre Channel Arbitrated Loop. This diagram serves only to provide an example of an application of a FC-analyser and should in no way be construed as limiting
15 the scope of the invention. Figure 4 should be viewed in conjunction with figure 1 to observe how the simplified representation of the analysis of a FC-AL shown in figure 4, relates to the integrated data gathering system for a FC-AL shown in figure 1. Figure 4 should also be viewed in
20 conjunction with figure 5 to observe how the logical connections between the devices on the FC-AL shown in figure 4 are physically implemented in a cross-point switch (30).

Looking initially at figure 4, there are shown two FC-AL
25 analysers, namely FC-AL Analyser 0 (150) and FC-AL Analyser 1 (152) corresponding to either the analysers 70 or 70' in Figure 1. The analysers are connected to the FC-AL via branches from the loop at points (154) and (156) respectively. The FC-AL has five nodes therein, of which three are hard
30 disks (Disk 0 (158), Disk 1 (160) and Disk 2 (162)). Of the remaining nodes one is a host CPU, Host A (164), with an AL_PA of 42 and the other is a repeater (166).

The purpose of a repeater is as follows; while it is possible to transmit signals for considerable distances over coaxial cable without degradation, differences in impedance across connections between components leads to degradation of a

5   signal and the necessity for repeaters to filter and amplify the signal. Since the disks in a FC-AL receive and actively transmit signals through their ports they effectively act as repeaters themselves. However, it is necessary to space the disks evenly about the FC-AL to achieve balanced signal

10 repetition. In a FC-AL with few disks, it is necessary to supplement the repeating activity of the disks by means of additional repeaters. However, whilst repeaters act to improve the quality of a transmitted signal, they have the disadvantage of adding to the latency of the loop. Taking

15 these two issues into account, the cross-point switch as will be described in figure 5, provides the facility for user-configurable or automatic, arrangement and use of repeaters, in order to optimise the performance of the FC-AL.

20 As can be seen from above, a repeater basically takes the fibre channel signal and cleans up the edges but does not alter the timing. A retimer takes the signal in it's serial form, extracts the clock with a PLL (phase locked loop) and retransmits the data synchronised to a new, externally

25 provided clock, thus removing jitter. A disk is a retimer, and both repeaters and retimers are available as standalone devices or embedded in other devices such as port bypass circuits.

30 Returning to figure 4, the two FC-AL analysers, (FC-AL Analyser 0 (150) and FC-AL Analyser 1 (152)) sample data from the FC-AL through their connection points (154 and 156). The data sampled from the FC-AL, by the two FC-AL analysers is

shown in the diagram as *Serial Data to Analyser* (through connections C6 and B7 (32 and 34)).

The two FC-AL analysers, (FC-AL Analyser 0 (150) and FC-AL

5   Analyser 1 (152)) are each equipped with a SCSI Enclosure Services (SES) Processor Interface (172 and 174 respectively). The SES Processor Interface (172 and 174) enables bi-directional communication between the FC-AL Analyser (150 and 152) and the SES Processor (7). Such bi-directional

10 communications are comprised of configuration commands sent to a given FC-AL Analyser from the SES processor (7) (shown in the diagram as *Analyser_Control_Signal*) and performance-related data transmitted from the FC-AL analyser to the SES processor (7) (shown in the diagram as *Analyser_Data*).

15 Communications between the FC-AL analysers (150 and 152) and the SES processor (7) are conducted through respective ESI busses (40 and 42). The methods for configuring the FC-AL analysers (150 and 152) will be described in further detail later in this section.

20

Figure 4 also shows as an example, an *ARB* ordered set (184) transmitted from Host A (164) to the next node on the FC-AL, namely a Disk 0 (158). The presence of the *ARB ordered set* (184) indicates that Host A (164) desires to gain control of

25 the FC-AL as described earlier.

Whilst a single FC-AL analyser (150 or 152) provides very detailed information concerning activity on the FC-AL at its connection point (154 or 156), the particular benefits of the

30 embodiment become more evident on comparing the data from a multiplicity of such analysers.

In figure 4, a FC-AL with two FC-AL analysers (FC-AL Analyser 0 (150) and FC-AL Analyser 1 (152)) is shown. If, for example, on comparing the number *of LIP ordered sets* detected by both analysers, it is found that the number of *LIP ordered*
5 *sets* detected by FC-AL Analyser 0 (150) is greater than that detected by FC-AL Analyser 1 (152), then such would indicate that Disk 2 (162) is likely to be a source of *LIP ordered sets*. Such in turn would indicate that Disk 2 (162) was out of synchronisation with respect to the rest of the components
10 on the FC-AL.

Figure 4 shows the logical connections between the devices in the specific example described above, however, figure 5 shows how these logical connections are implemented physically by
15 means of a cross-point switch.

A cross-point switch (or cross-bar switch) (CPS) comprises a matrix of switches connected by signal lines, thereby creating a switching device with a fixed number of inputs and outputs.
20 A CPS (30) can be constructed according to one of the following architectures:
(i) Concentration : more input lines than output lines
(ii) Expansion : more output lines than input lines
(iii) Connection : an equal number of input and output lines
25

In the example given in figure 5, a CPS (30) with connection architecture (a square matrix of switches) is employed with 8 inputs and 8 outputs. It must be emphasised once again, that this diagram serves only as an example of an implementation of
30 the CPS (30) and should in no way be considered as limiting the scope of the invention. The inputs to the CPS (30) are located on the left-hand side of the square matrix and are labelled with letters A to H from the top down. The outputs

from the CPS (30) are located at the bottom of the square matrix and are labelled 0 to 9 running from left to right. At the intersection of each input and output line, there is provided a switched connection which, for the purposes of the present description, will be labelled with the letter and number of the input and output lines between which the switched connection can make or break a circuit. In figure 5, closed connections (switches) are shown as solid circles and open switches are shown as hashed circles. Solid lines are used to indicate a signal transmitted from a connected input device to a connected output device, whereas unused CPS input and output lines are shown as shaded lines.

A range of devices are connected to the inputs and outputs of the CPS (30), these devices correspond to the devices described earlier in relation to figure 4. The output of Disk 0 is connected to CPS input A, the output of Disk 1 is connected to CPS input B and the output of Disk 2 is connected to CPS input C. The output of the repeater (166) is connected to CPS input E and the output of Host A is connected to CPS input F, the other inputs to the CPS (D, G and H) remain unconnected.

Disk 1 receives its input from CPS output 0 via CPS connection A0 and Disk 2 receives its input from CPS output 1 via CPS connection B1. Further, the repeater (166) and Host A receive their inputs from CPS outputs 2 and 3 respectively, via CPS connections C2 and E3 respectively. Disk 0, FC-Analyser 0 (150) and FC-Analyser 1 (152) receive their inputs from CPS outputs 4, 6 and 7 via CPS connections F4, C6 (32 in Figure 4) and B7 (34 in Figure 4) respectively. The logical links in the FC-AL depicted in figure 4 are shown with the

corresponding alphanumeric designation from the CPS connection
loops shown in figure 5.

Looking at the FC-AL in figure 4, it can be seen that Host A
5   (164) is logically connected to Disk 0 (158). This
association is physically implemented in figure 5 by
connecting the output from Host A on CPS input line F to CPS
output line 4 through the fifth switch on CPS input line F.
Similarly the logical connection between Disk 0 and Disk 1 in
10   figure 4 is physically implemented in figure 5 by connecting
the output from Disk 0 on CPS input line A to CPS output line
0 through the first switch on CPS input line A.

It can also be seen in figure 4 that Disk 1 is connected both
15   to Disk 2 and FC-Analyser 1 (152). However, whilst Disk 2 is
logically an element in the FC-AL, the FC-Analyser 1 (152)
samples data from the FC-AL on a branching connection
therefrom, without itself contributing to the latency of the
FC-AL. Such connection structure is physically implemented in
20   figure 5 by connecting the output voltage signal from Disk 1
on CPS input line B to the CPS output lines 1 and 7 through
the second and eighth switches on the CPS input line B. The
FC-Analyser 1 (152) is connected to the CPS output line 7
through the CPS connection loop B7 (34) and the Disk 2 is
25   connected to the CPS output line 1 through the CPS connection
loop B1 (188). However, whilst Disk 2 continues the FC-AL by
transmitting its output to CPS input line C, the FC-Analyser 1
(152) transmits the results of its analyses directly to the
SES processor (7) and thereby does not itself contribute to
30   the loop delay on the FC-AL. From the SES processor (7), the
results of the FC-AL analysis are processed and transmitted to
a disk on the FC-AL via the Data Gatherer Chip (50) (shown in

figure 1 but not in figure 4) through SPI and ESI busses ((54) and (52) in figure 1).

For the sake of brevity, the physical connections between the
5   remaining nodes and FC-analyser for the FC-AL depicted in figure 4 will not be described here, but can be ascertained on examination of figure 5.

The SES processor (7) is also in bi-directional communication
10   with any FC-analysers (70) (via 40 and/or 42) and unidirectional communication with the CPS (30) (via 36). The bi-directional link between the SES processor (7) and the host CPU (82), enables the SES processor (7) to transmit the results of any environmental monitoring or traffic analysis
15   from the FC-AL analyser (70) to the host CPU (82). However, the bi-directional link also enables the host CPU (82) to issue configuration commands to the SES processor (7), which the SES processor (7) in turn transmits to the CPS (30) and/or the FC-AL analyser (70).

20

The communication links between the host CPU (82) and the FC-AL analyser (70) via the SES processor (7) allows the FC-AL analyser (70) to be programmed by the user to measure particular analytical variables relating to the performance of
25   the FC-AL. Such user-configurable data acquisition is enabled by software, running on the host CPU (82) (for example Vision, further details available at http://www.eurologic.ie/products/vision.htm), which packets the configuration requirements of the user into a form that
30   can be interpreted by the SES processor (7) (e.g. configuration pages). On receiving this information the SES processor (7) determines the appropriate destination for the

configuration commands and transmits it to the destination in the appropriate form.

Similarly, information from the FC-AL analyser (70) is
5   transmitted to the SES processor (7) as, for example, a status page, and thence to Vision (or other similar software) on the host CPU (82) and displayed to the user in a more accessible format.

10   Communication between the host CPU (82) and the CPS (30) via the SES processor (7) allows the configuration of the on/off states of the different switches in the CPS matrix (30) to be programmed by the user. Such configuration of the CPS (30) thereby determines the connection sequence of nodes in the FC-
15   AL and the structure and placement of the branching connections for FC-Analysers on the FC-AL. Configuration of the CPS (30) by the user is also enabled by software running on the host CPU (82) (for example Vision as described above).

20   Thus, on start-up, the system operates in an autonomous mode wherein any disks connected to the storage rack transmit a signal to the Data Gatherer Chip (50) on the $Pres_{1-m}$ lines, Figure 1. This signal notifies the Data Gatherer Chip (50) of the presence of the connected disks and the SES processor (7)
25   in turn obtains this information from the Data Gatherer Chip (50). (Alternatively, if data gatherer chips 50,50' are not employed, the SES Processors could receive the present inputs P directly from the disks, although this would increase the cost of the edge connector required to bring the signals onto
30   the SES processor board (4).) The SES Processor then informs the CPS (30) to make the appropriate connections to form the loop between the disks and host(s), and once the loop has been established, it is then possible for a user to configure the

CPS (30) as desired both to re-order devices within the loop and to select the points at which the analysers connect to the loop.

5   The advantages of this method of connecting the FC-AL analyser (70) and FC-AL nodes via the CPS (30) is that firstly it is possible to for the user to selectively place the FC-AL analyser (70) on the FC-AL loop without contributing to the latency of the FC-AL. Whilst the process of reporting the

10   results of the FC-AL analyser's analyses contributes to the traffic on the FC-AL, such contribution constitutes approximately 0.1% of the bandwidth of the FC-AL and as such is not significantly detrimental to the performance of the FC-AL.

15

       Secondly, it is possible for the user to re-order the connections between the different FC-AL nodes a facility that is not available with a conventional port bypass circuit.

20   Figure 6 shows a more detailed block diagram of the FC-AL analyser (70) itself. Serial data on the FC-AL (shown in figure 4 as *Serial Data to Analyser* (32)) is transmitted to a serialiser-deserialiser (SERDES) (244). In the embodiment shown, the SERDES (244) employed is a Vitesse 7126. However,

25   it should be recognised that the scope of the invention is not limited to a particular SERDES (244). The SERDES (244) samples the received serial data (32). The sampled data is re-timed by the SERDES (244) according to an internal clock. The internal clock is phase-locked to the received serial data

30   (32) (further details can be obtained from Vitesse Data Sheet VSC7126).

The SERDES (244) has two outputs in this embodiment. To generate the first output, the re-timed data is deserialised into two 10-bit characters. The two 10 bit characters are concatenated to form a 20 bit character and output onto a 20 bit data bus as *Deser_FC-AL_Data* (246).

To generate the second output, the SERDES (244) detects FC comma characters in the sampled serial data (32). The detected comma is output on a separate bus from the deserialised data as *FC-AL_Status_Data*. (252). The *FC-AL_Status_Data* (252) is stored in a status register (254) and output as *FC-AL_Status* (256). The component of the embodiment to which this data is transmitted will be discussed later in the description.

Returning to the *Deser_FC-AL_Data* (246), consecutive characters on the 20 bit wide bus are stored in one of two data registers, namely FC_AL Data Register 0 (258) and FC-AL Data Register 1 (260). The FC-AL Data Register 0 (258) and FC-AL Data Register 1 (260) each have another input, namely control signals on a *Load_Reg_0* (248) line and a *Load_Reg_1* (250) line respectively. Such signals act to enable and disable the ability of a given register to accept an input. Such signals thereby determine to which of the two registers a given character from the *Deser_FC-AL_Data* (246) is transmitted. However, in this embodiment the FC-AL Data Register 0 (258) and FC-AL Data Register 1 (260) take alternate turns in accepting characters from the *Deser_FC-AL_Data* (246).

The FC-AL Data Register 0 (258) and FC-AL Data Register 1 (260) have one output each, along which they output their 20 bit characters as *FC-AL Coded_Data_0* (262) and *FC-AL*

Coded_Data_1 (264) respectively. These two outputs are transmitted together to two, separate detection modules, namely an ordered set detection module and a 10B/8B decoding module.

5

Looking at the first of these modules, namely the ordered set detection module, this module is shown as an *Ordered_Set_Detect* block (266) in figure 6. This module serves to detect *ordered sets* in data sampled from the FC-AL.

10 The ordered set detection module (266) also performs run-length checking.

Whilst the *Ordered_Set_Detect* block (266) supports the detection of a pre-defined set of commonly occurring *ordered sets*, it is also a user programmable component, enabling the

15 user to specify particular *ordered sets* to be detected. Such configuration commands are transmitted to the *Ordered_Set_Detect* block (266) by the SES Processor (7) via the SES Processor Interface (240). The configuration commands

20 are depicted in figure 5 as an *Analyser_Control_Signal* (242).

The *Ordered_Set_Detect* block (266) also has as an input, the output signal from the status register (254) namely the *FC-AL_Status* (256). Such input enables the *Ordered_Set_Detect*

25 block (266) to serve as a means of status checking and K28.5 detection.

Having detected and identified specific ordered sets, the *Ordered_Set_Detect* block (266) produces three outputs, namely,

30 *SOF* (268), *EOF* (270) and *Filtered_Ordered_Sets* (272). Looking at the first two of these outputs (i.e. *SOF* (268) and *EOF*(270)), the *Start_of_Frame (SOF) ordered set* (112) and *End_of_Frame (EOF) ordered set* (120) are isolated from a given

set of *ordered sets* which had been detected and identified by the *Ordered_Set_Detect* block (266). The isolated *ordered sets* are then transmitted to a CRC Verification Block (282) along the *SOF* (268) line for the *Start_of_Frame ordered sets* (112)

5   and *EOF* (270) line for the *End_of_Frame ordered sets* (120). This description will return to the CRC Verification Block (282) later.

We return now to the third output from the *Ordered_Set_Detect*

10   block (266), namely the *Filtered_Ordered_Sets* (272). Following the isolation of the *Start_of_Frame (SOF)* and *End_of_Frame (EOF) ordered sets*, the *Filtered_Ordered_Sets* (272) output is used to transmit the remaining *ordered sets* detected and identified by the *Ordered_Set_Detect* block (266)

15   to a set of ordered set counters (274). The ordered set counters (274) will be described later in the description.

Turning now to the second module to which the FC-AL Data Register 0 (258) and FC-AL Data Register 1 (260) transmit

20   their outputs, this is shown in the diagram as a module comprised of four 10B/8B Decoding blocks (276). The purpose of the 10B/8B Decoding blocks (276) is to decode the 40 bits characters received from the FC-AL Data Registers (258 and 260) (i.e. *FC-AL Coded_Data 0* (262) *and FC-AL Coded Data 1*

25   (264)) into 32 bit characters. Such decoding is performed in accordance with the inverse of the 8B/10B encoding scheme described earlier. The resulting 32 bit characters are output from the 10B/8B decoding blocks along a single bus (shown as *FC-AL Decoded_Data* (278) in figure 6) to two further modules,

30   namely the CRC Verification Block (282) and a Frame Detection Block (280)). Each of these modules will be discussed in greater detail later in the description.

Returning to the ordered set counters (274), the LIP counters
(284), ARB counters (286) enumerate the number of occurrences
of these common *ordered sets* over a period of time. Further,
in correspondence with the facility for user-programmable,

5 specific *ordered set* detection provided by the
Ordered_Set_Detect block (266) as described above, the ordered
set counters (274) also count the occurrences of the user
specified *ordered sets*. Such counters are depicted as OS
counters x0-xn (288) in figure 6.

10

The ordered set counters (274) will also count the number of
occurrences of Run Length Disparities (RLDs) in the RLD
counter (290). An RLD is used as an indicator of lack of
synchronisation but is not strictly an *ordered set*. The

15 ordered set counters (274) have one output which is
transmitted to the SES processor (7) via the SES processor
interface (240).

Returning now to the CRC Verification Block (282), it will be
20 recalled that this block has three inputs, *SOF* (268), *EOF*
(270) and *FC-AL Decoded_Data* (278). The CRC Verification
Block (282) uses the information from the CRC (118) part of
the frame so delimited, to enable error detection in the
associated frame. The validity or invalidity of a frame as
25 detected by the CRC Verification Block (282) is flagged as
such by the CRC Verification Block (282) and output as a
Frame_Validity_Flag (292). The *Frame_Validity_Flag* (292) is
transmitted to two separate modules namely the Frame Detection
Block (280) and a block of counters for the number of
30 occurrences of valid and invalid frames (294) over a period of
time. The data from the number of valid frames and number of
invalid frames counters (294) are output to the SES processor
(7) via the SES processor interface (240).

Returning to the Frame Detection Block (280), it will be recalled that this block has two inputs, namely the *Frame_Validity_Flag* (292) and the *FC-AL Decoded Data* (278).
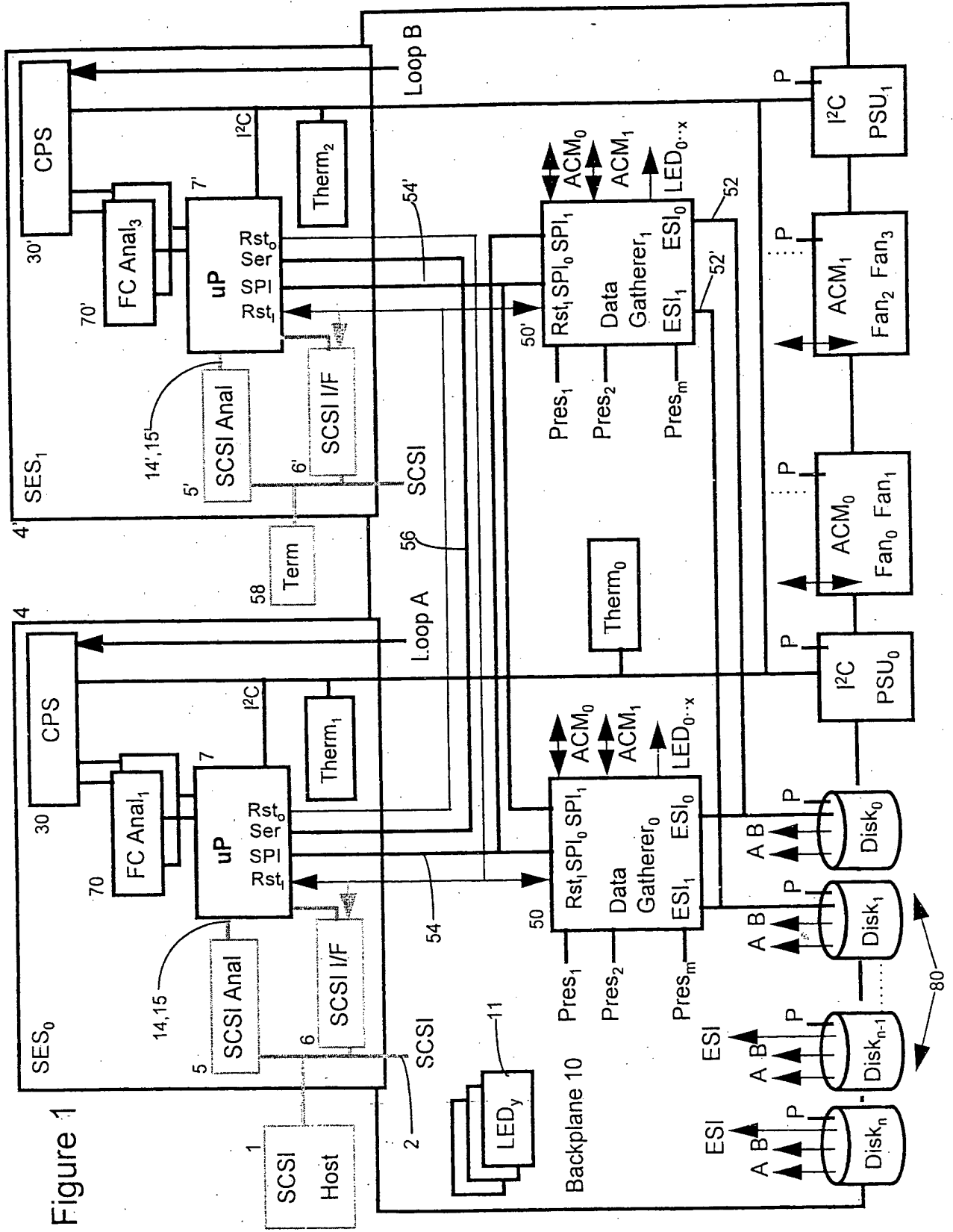
5 The Frame Detection Block (280) isolates header information such as source address, destination address etc. from a frame. If the CRC Verification Block (282) flags that the associated frame was invalid via the *Frame_Validity_Flag* (292) signal, then the information isolated in the Frame Detection Block

10 (280) is transmitted to a block of registers, namely the Last Bad Frame Data Registers (296) as *Assessed_Frames_Data* (298). In the Last Bad Frame Data Registers (296) individual isolated frame attributes are written to their corresponding register (e.g. source address etc.). However, if the CRC Verification

15 Block (282) flags that the associated frame was valid, then the information isolated in the Frame Detection Block (280) is discarded.

The data contained in the Last Bad Frame Data Registers (296)

20 are output to the SES processor (7) via the SES processor interface (240). However, if the data received by the FC-AL analyser is of very poor quality (i.e. with a high rate of invalid frames) it is possible that received frames may be recognised as invalid faster than it is possible for the SES

25 processor (7) to read the data from the Last Bad Frame Data Registers (296). In such circumstance, the number of invalid frames counter (294) will continue incrementing itself in response to the recognised bad frames. However, in order to reduce the risk of overwriting data in the Last Bad Frame Data

30 Registers (296), the Last Bad Frame Data Registers (296) are such that it is not possible to write more information to them until their current contents have been read by the SES processor (7). For example the Last Bad Frame Data Registers

(296) comprises a sample and hold component, with a sampling
rate matching the rate at which the SES processor (7) can
recover the data from the Last Bad Frame Data Registers (296).
Hence if frames are being recognised as being invalid faster

5   than the SES processor (7) can read the header data from the
Last Bad Frame Data Registers (296), the Last Bad Frame Data
Registers (296) will only hold data from the last invalid
frame detected by the CRC Verification Block (282).

10   In summary, the FC-AL analyser accepts as input, data from the
FC-AL and configuration commands from the SES processor (7)
and outputs to the SES processor (7) the information from the
ordered set counters (274), the Last Bad Frame Data Registers
(296) and the number of valid and invalid frames counters

15   (294).

Where the SES processor (7) receives information from more
than one analyser (70) on a loop, it can then collate this
information and even make a diagnosis of a problem on the

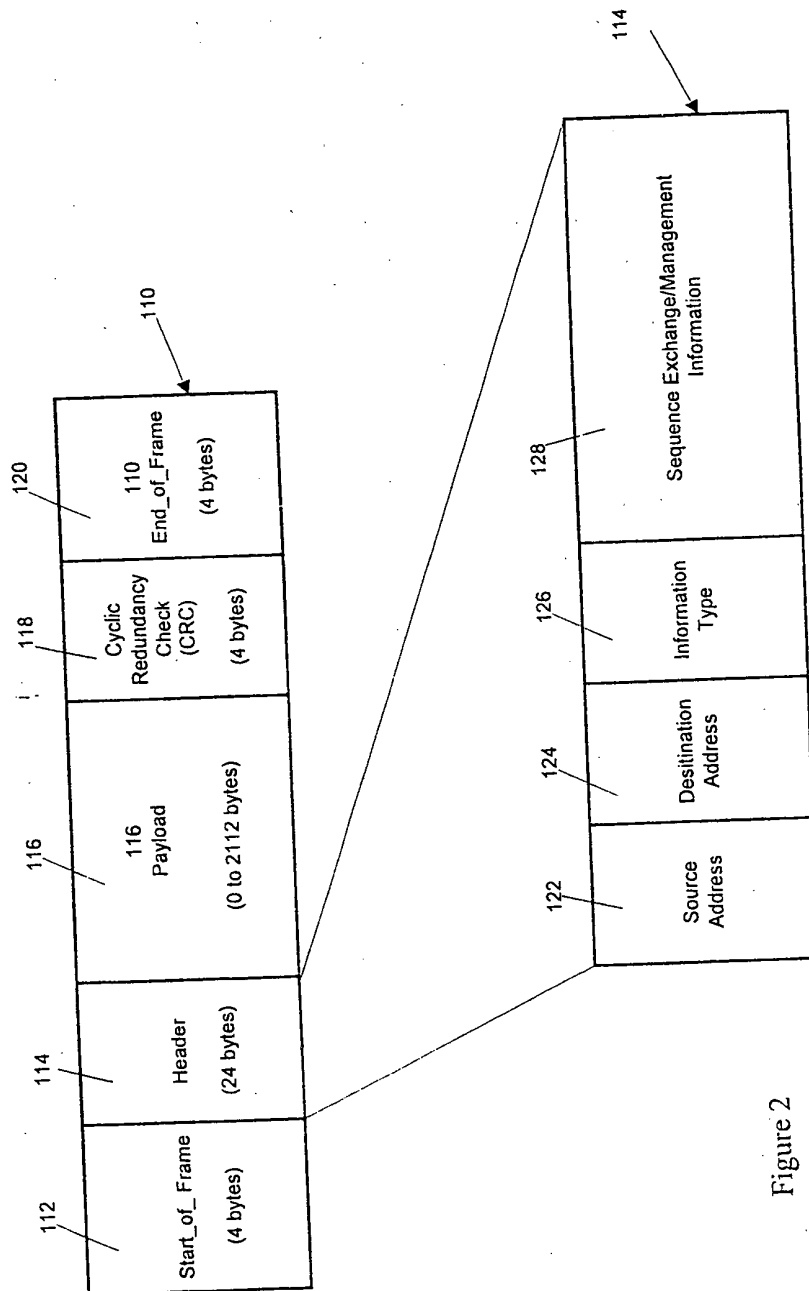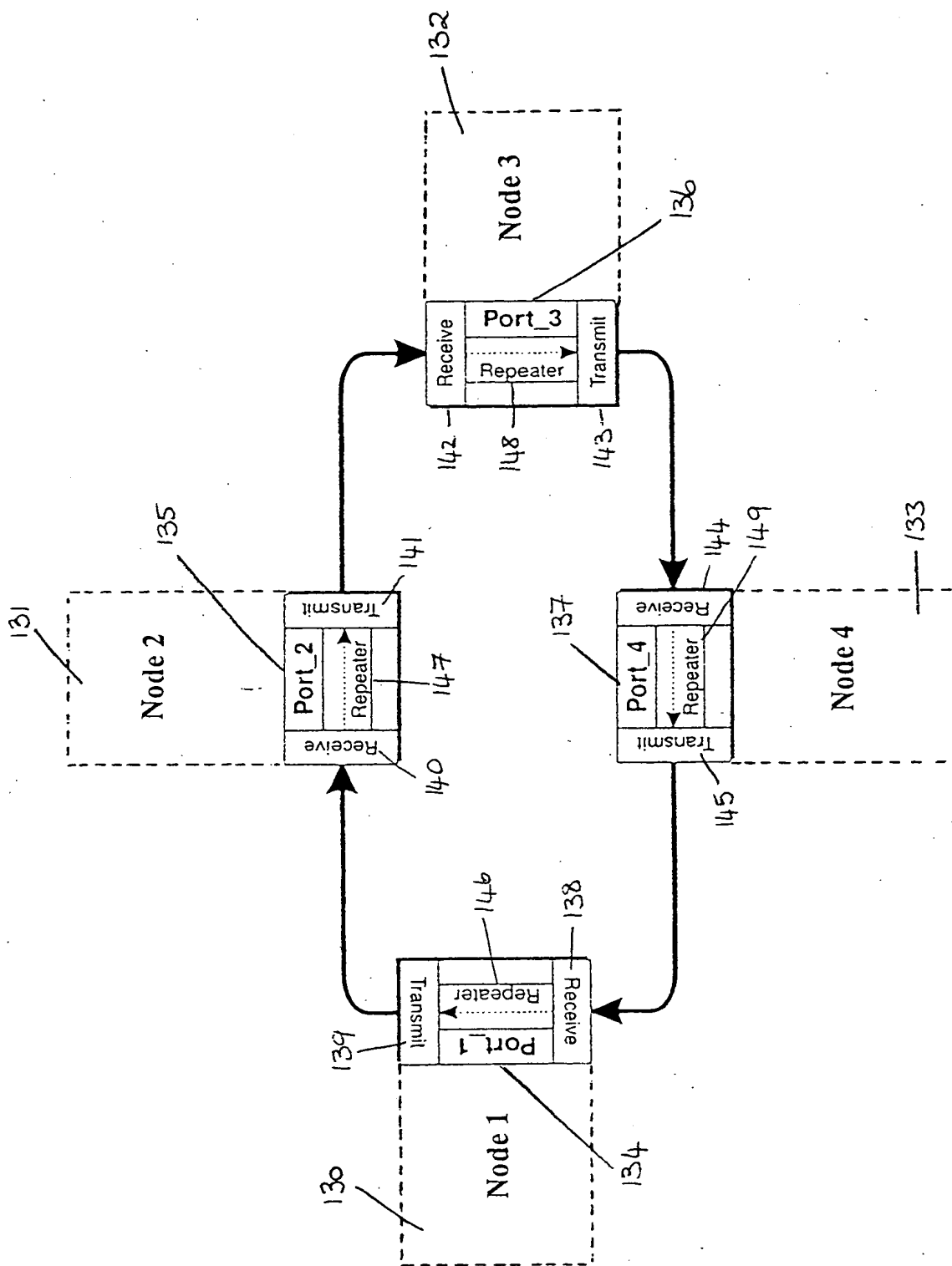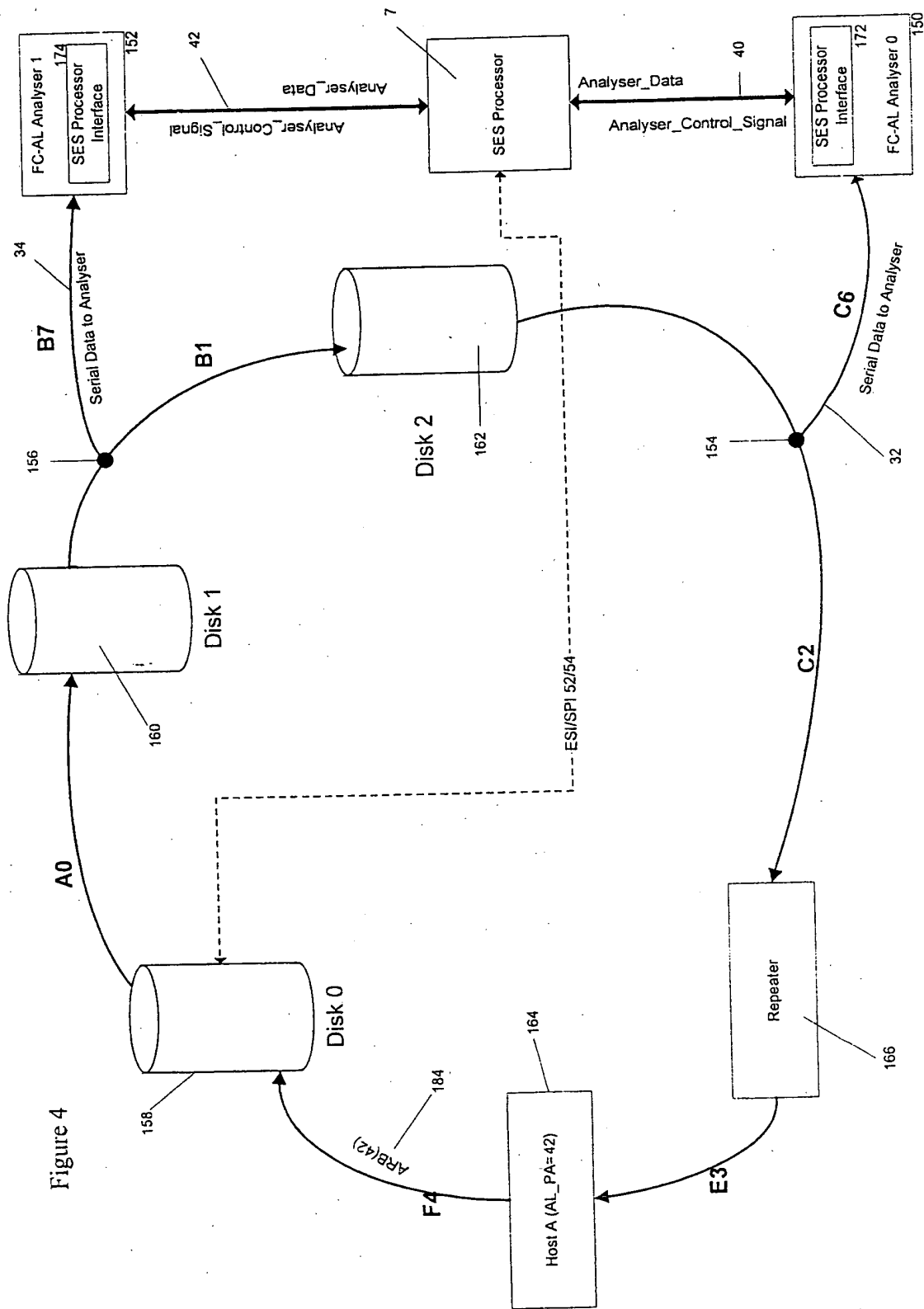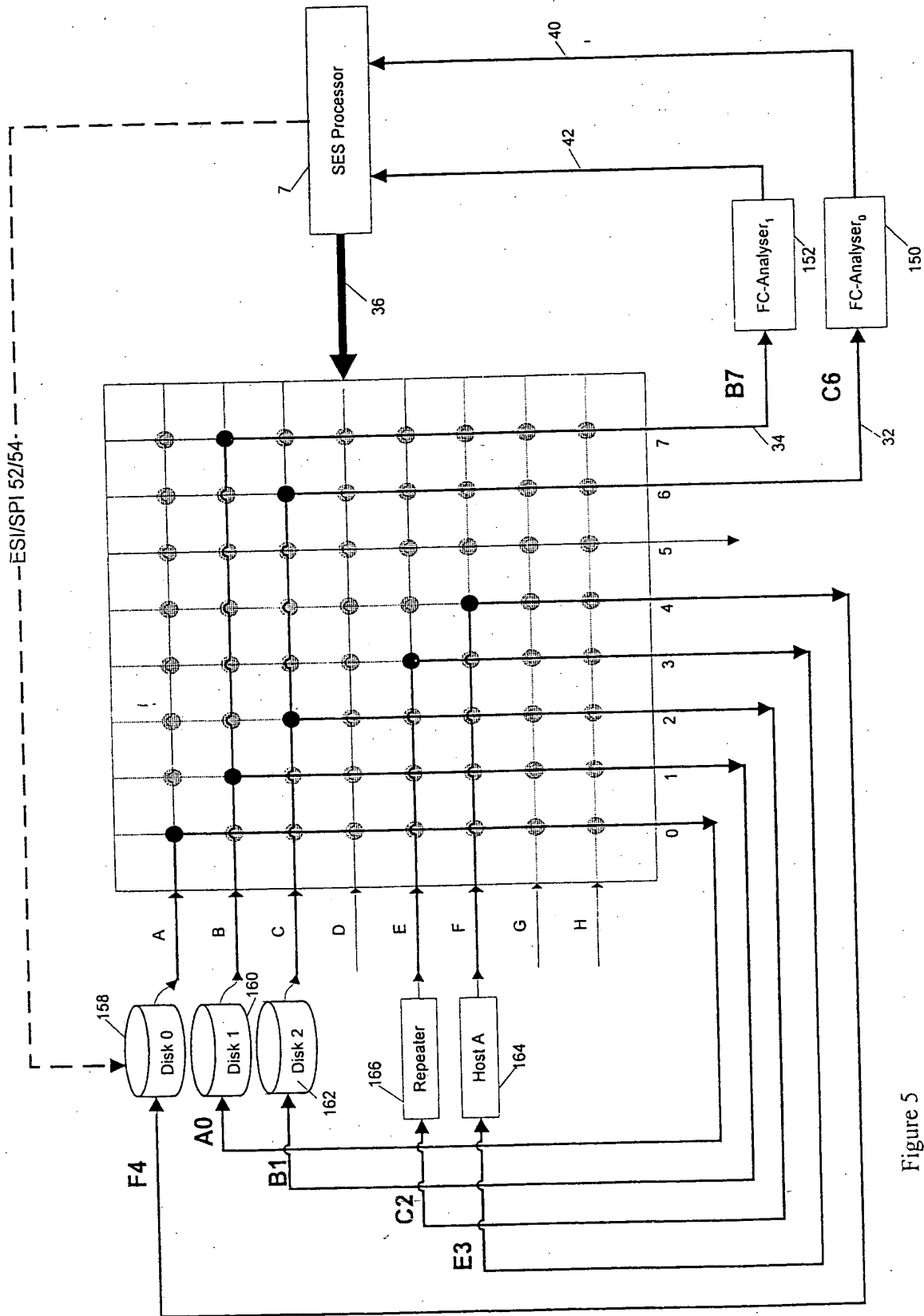20   loop, before reporting this problem to a host application.

Figure 1

Figure 2

Fig. 3

Figure 4

Figure 5

Figure 6